

## • 嵌入式系统 •

## 基于 S3C2410 的 WMA 开源解码程序优化

张海滨, 李 挥

(北京大学 深圳研究生院集成微系统科学与工程应用重点实验室, 广东 深圳 518055)

**摘 要:** 针对 MPlayer 中 WMA 解码程序在 S3C2410 平台上解码效率低的问题, 根据 S3C2410 平台的特性提出并分析了浮点运算转定点运算、以空间换时间以及手工汇编 3 种优化方案。详细介绍了定点化原理, 并提出了模块化的优化方式, 对 MPlayer 中的 WMA 解码程序进行优化。实验的结果表明了优化后的 WMA 解码程序在 S3C2410 平台上的解码效率已得到显著提高, 已经能够流畅播放单通道的 WMA 文件。

**关键词:** 嵌入式系统; 浮点转定点; MPlayer; WMA 解码优化; S3C2410

中图分类号: TP311.54 文献标识码: A 文章编号: 1000-7024 (2009) 01-0013-03

## Optimizing open source decoding program of WMA based on S3C2410

ZHANG Hai-bin, LI Hui

(Key Laboratory of Integrated Microsystem Science and Engineering Applications, Shenzhen Graduate School, Peking University, Shenzhen 518055, China)

**Abstract:** For the problem of low efficiency of WMA (Windows Media Audio) decoder, which within MPlayer, on the platform of S3C2410. Three optimizing methods according to the characters of S3C2410 are analyzed. The methods include converting float point number to fixed point number, using space to save time and using assemble language. Moreover, the principle of fixing float number is introduced in details. Then modular method for optimizing decoding program of WMA is proposed. Eventually, it shows great progress in efficiency of WMA decoder on S3C2410 through comparative experiment. And it can decode single channel WMA file with great fluency.

**Key words:** embedded system; float number to fixed number; mplayer; optimizing WMA decoder; S3C2410

## 0 引言

WMA 作为一种与 MP3 格式齐名的音频压缩格式, 以其良好的特性在数字电台架设、在线试听等领域得到了广泛的应用。除了微软的解码程序外, MPlayer 提供了开源的 WMA 解码程序。但 MPlayer 中的 WMA 解码程序运算量大, 移植到一些运算能力有限的嵌入式平台后, 解码效率未能满足要求。另外, ARM 公司推出的一系列 ARM 处理器则占据了全球嵌入式芯片市场的大量份额, S3C2410 就是一款具有代表性的 ARM 内核处理器。本文以 S3C2410 为目标平台, 对 MPlayer 中的 WMA 开源解码程序进行移植和优化, 不仅能提高 WMA 解码程序在 S3C2410 中的解码效率得到显著提高, 也能对嵌入式开发中类似的优化工作起到借鉴作用。

## 1 目标平台简介

在优化过程中, 我们采用的目标平台是 fft-2410 实验箱, 它采用 S3C2410 处理器和韩国 MIZI 公司提供的开源嵌入式操作系统 arm-Linux。S3C2410 是三星公司基于 ARM920T 处

理器内核开发的一款 16/32 位嵌入式处理器, 内部带有全性能的 MMU, 具有高性能、低功耗、体积小和接口丰富等优良特性<sup>[1]</sup>, 但 S3C2410 不具有 FPU(浮点运算单元)。

MPlayer(我们采用 MPlayer-1.0rc1 版本<sup>[2]</sup>)是一个支持多平台的开源媒体播放器, 通过调用各种编解码库, 能够播放大多数音视频文件, 应用广泛<sup>[3]</sup>。MPlayer 通过调用 FFmpeg 的 libavcodec 编解码库来解码 WMA 文件。解码时, MPlayer 主程序调用 WMA 解码程序的过程如图 1 所示。

解码函数主要由 wma\_decode\_superframe、wma\_decode\_

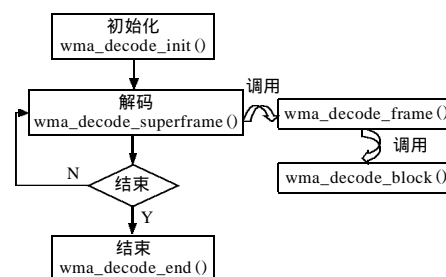


图 1 WMA 主要解码函数及其关系

收稿日期: 2008-07-07 E-mail: hbzhang2003@gmail.com

基金项目: 国家 863 高技术研究发展计划基金项目 (2007AA01Z218)。

作者简介: 张海滨 (1984 - ), 男, 福建南靖人, 硕士研究生, 研究方向为软件工程、嵌入式系统; 李挥 (1964 - ), 男, 广东汕头人, 博士, 副教授, 博士生导师, 研究方向为通信系统及芯片手机、嵌入式系统等。

frame和wma\_decode\_block这3个函数组成。这些函数在WMA解码过程中被频繁调用,耗时较多,需要优先对其进行优化。

## 2 优化方案的提出与分析

把MPlayer源码通过交叉编译,成功移植到S3C2410平台后,尝试使用MPlayer来播放WMA文件,可以看到如下现象:初始化过程慢,播放时声音断断续续,而且停顿时间很长,不能流畅播放WMA音频文件。另外,MPlayer在播放时会在终端上实时显示当前CPU使用率,此时显示的CPU使用率为1000%左右。这表明了MPlayer中的WMA解码库在S3C2410上的解码速度不够快,需要对其进行优化,以提高解码速度。

通过分析WMA解码程序并结合S3C2410的特性,提出了以下3种优化方案来进行优化:浮点运算转定点运算、以空间换时间、手工汇编优化。

### 2.1 浮点运算转定点运算

在计算机内部表示数据时,小数点位置既可以是固定的,也可以是变化的。如果小数点的位置是固定的,此类型的数据称为定点数,例如C语言中的int类型。如果小数点位置是可变的,则称为浮点数,例如C语言中的float类型<sup>[4]</sup>。

由于浮点数的特殊存储格式,在进行浮点运算时,需要对浮点数进行对齐、运算和规范化等操作。所以对于不具有FPU的平台来说,需要通过软件模拟来实现浮点运算。这种方式运算量大,效率不高。前面已经提到,S3C2410不具有FPU,而WMA解码程序又涉及大量的浮点运算,对浮点运算速度要求较高。因此,把浮点运算转为定点运算是一种可行的优化方案,具有较大优化空间。

#### 2.1.1 定点化原理

浮点运算转定点运算也称为定点化,其原理是:把定点数存储空间的一部分用来表示浮点数的整数部分,剩余部分用来表示小数部分,从而替换掉浮点数进行定点运算,并对运算结果进行调整以得到正确结果的过程。

先看一个简单的例子:用16位short类型的定点数来代替浮点数,0~7位表示浮点数的小数部分,8~14位表示整数部分,15位表示符号。则虚拟的“小数点”位置就固定在第7位和第8位之间。按这种定点数格式替换浮点数,十进制数1.25的表示情况如图2所示。它表示的整数值为320,相当于把1.25乘以 $2^8$ 。

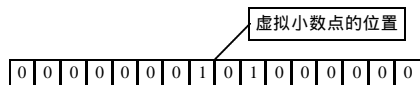


图2 十进制数1.25的定点表示

为了方便,我们用 $N$ 表示有符号定点数的位数, $f$ 表示小数部分所占的位数, $i$ 表示整数部分所占的位数。把具有这种格式的定点数记为 $F(i,f)$ (下同)。设原来的浮点数为 $r$ ,由于定点数可以由相应的浮点数乘以 $2^f$ 而得到<sup>[5]</sup>,则有

$$F(i,f) = r \times 2^f \quad (1)$$

设 $x_n$ 表示 $F(i,f)$ 中第 $n$ 位的值,根据每一位的权重求和有<sup>[6]</sup>

$$F(i,f) = (1/2^f) [-2^{N-1} x_{N-1} + \sum_{n=0}^{N-2} 2^n x_n] \quad (2)$$

其中, $N = i + f + 1$ 。

由式(2)可以看出定点数 $F(i,f)$ 所能表示的浮点数范围为: $-2^i F(i,f) \sim 2^i - 1/2^f$ 。因为 $1/2^f$ 相对于 $2^i$ 来说是一个较小的数,所以通常近似地认为 $F(i,f)$ 所能表示的浮点数范围是由 $i$ 决定的。另外, $F(i,f)$ 的分辨率<sup>[6]</sup>(Resolution,即能表示的最小数量级,相当于最小刻度)为 $1/2^f$ 。因此, $F(i,f)$ 所能表示的浮点数精度则由 $f$ 决定。由 $N = i + f + 1$ 可知,当 $N, i, f$ 确定了两个以后,另外一个也就确定了。定点数一般采用高级语言中的整数类型,所以 $N$ 常取值为16,32,64。 $N$ 值确定后,当 $f$ 取值越大,精度就越高,而 $i$ 值则越小,取值范围也越小。反之,当 $f$ 取值越小,精度就越低,而 $i$ 值则越大,取值范围也越大。

解决了定点数替代浮点数问题后,还需要对定点运算结果做适当调整才能得到正确的结果<sup>[5]</sup>。

(1)对于定点数加法和减法,如果两个定点数的虚拟小数点的位置不一致,需要通过移位使两个数先对齐,再进行运算。由式(1)可知,对齐后运算公式如下

$$F_1(i,f) \pm F_2(i,f) = r_1 2^f \pm r_2 2^f = (r_1 \pm r_2) 2^f = F_3(i,f) \quad (3)$$

由式(3)可以看出,结果与操作数的格式是一样的,不需要调整。

(2)对于定点数乘法,不需要先对齐而需要对结果进行移位调整。如果两个操作数对齐,由式(1)可知,定点乘法运算公式如下

$$F_1(i,f) \times F_2(i,f) = r_1 2^f \times r_2 2^f = (r_1 \times r_2) 2^{2f} = F_3(i,f) \times 2^f \quad (4)$$

由式(4)可以看出,需要把结果除以 $2^f$ (右移 $f$ 位)才能得到正确结果。如果两个操作数没有对齐,原理也一样,需要对结果进行移位调整才能得到正确结果。具体移位操作要根据源操作数以及结果所需的定点数格式来确定。

(3)对于定点数除法,不需要先对齐而需要对结果进行移位调整。如果两个操作数对齐,由式(1)可知,定点除法运算公式如下

$$F_1(i,f)/F_2(i,f) = r_1 2^f / r_2 2^f = r_1 / r_2 = F_3(i,f) / 2^f \quad (5)$$

由式(5)可以看出,需要把结果乘以 $2^f$ (左移 $f$ 位)才能得到正确结果。这种先进行除法再移位的方式会使精度大大损失,因此通常是采取先对被除数移位再进行除法运算的方式来减少精度损失,式(5)可变换为

$$F_1(i,f) \times 2^f / F_2(i,f) = r_1 2^{2f} / r_2 2^f = (r_1 / r_2) 2^f = F_3(i,f) \quad (6)$$

如果两个操作数没有对齐,情况也类似。

在做移位调整的过程中容易产生溢出以及精度损失,因此,在定点化过程中要特别注意这一点。而对于无符号定点数的原理也一样,这里就不再赘述。完全定点化后,就转换成只有整数参与的运算,这与软件模拟浮点运算的方式存在着本质区别。

#### 2.1.2 定点化步骤

根据上节的定点数学理论,在对应用程序中的浮点数变量(通常要把几个参与运算的变量同时定点化)进行定点化时,一般按照以下步骤来进行:

- (1)分析程序中浮点数变量所要求的精度和范围;
  - (2)根据定点化原理,确定 $N, i, f$ 这3个数并定义相应的定点数变量;
  - (3)用定点数替换浮点数,并对运算结果进行相应调整。
- 在优化过程中,大部分采用64位有符号整数,以 $F(39,24)$

这种定点数格式对 WMA 解码程序完成了定点化。

## 2.2 以空间换时间

目前,存储器的容量越来越大,价格也变得越来越便宜。因此,这就允许我们在适当的情况下牺牲部分存储空间来获得较快的运算速度,这就是所谓的以空间换时间。主要包含两种方式:一种是把一些被频繁调用而且比较短小的函数用内部代码来替换函数调用。如 C++ 中的 inline 关键字就能起到此作用。这样必然会增加代码的空间,但同时也减少了频繁调用函数的时间,从而提高代码的执行效率。另一种是用查表法来代替复杂的运算,即事先计算好复杂运算的值并存在表中,在运算时直接在表中查询。虽然需要花费空间来存储数值表,但可以省去大量的计算时间<sup>[7]</sup>。例如  $y = \sin(x)$ ,  $y$  为 float 类型,  $x$  为 int 类型,且  $0 \leq x \leq 100$ 。首先计算出  $\sin(x)$  ( $0 \leq x \leq 100$ ) 的值并存在数组中 `sin_table[100]` 中,然后用 `y = sin_table[x]` 替换 `y = sin(x)` 即可。如果需要定点化,也要把数值表定点化。

在 S3C2410 平台上,把 `sin()`、`cos()` 以及 `pow()` 等函数,用定点化的数值表来代替,可以有效地减少运算时间,大大提高运行效率。

## 2.3 手工汇编优化

汇编语言是面向机器的程序设计语言,能较好地发挥机器的特性,得到质量较高的程序,而且目标程序占用内存空间少,运行速度快,有着高级语言不可替代的功能。尽管大部分 C/C++ 编译器能够产生高质量的机器代码,但对一些关键运算采用手工汇编优化能够达到比编译器更好的效果<sup>[7]</sup>。这就要求程序员熟悉目标平台的架构以及指令系统,然后分析程序的算法以及结构,通过调整语句顺序和尽量使用寄存器等方式,以高效的指令实现程序功能。

我们在对 WMA 解码优化过程中,用 ARM 汇编实现了 64 位定点数乘法(包含溢出处理)等运算,相比 C 语言实现的 64 位定点数乘法运算,运算速度提高了 68% 左右。

## 3 以模块方式实现优化

对代码量大的程序进行修改,如果采取一次性修改完再测试的方式,一旦出现错误,就难于找出错误的地方。当代码量越大,程序算法越复杂时,这个问题也就越明显。因此,我们提出以模块化的方式来优化 WMA 解码程序,这里的模块可以是一个函数,也可以是程序里的一小部分,模块的分法视具体情况而定。当一个模块优化完后,就立刻进行测试,如果有错误,就只需要在该模块内查找错误即可;如果没有错误,就继续优化其他模块。模块化方式既简化了优化过程,也保证了优化结果的正确性。

以定点优化为例,如图 3 所示,两条横线之间代表一个模块。首先,在模块的入口处把模块中参与运算的浮点数转为定点数。接着,把浮点运算替换为定点运算。最后,把定点数结果转为浮点数。定点化后,就对该模块进行测试直到没有错误。之所以要使用 `floattofixed` 以及 `fixedtofloat` 这两个变换,是因为变量 `a` 和 `f` 在后续模块中还要以浮点数的形式参与运算。这类似于中断机制里的现场保护以及现场恢复。尽管优化后的代码看似更加复杂,但这只是暂时的,当你全部优化完毕

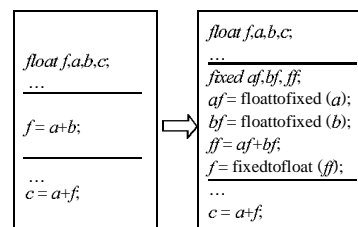


图 3 模块优化方式说明

后,这两个转换函数就可以消除掉。它们只是模块优化方法中的两个辅助函数而已。

## 4 优化结果分析

通过上述 3 种方案对 WMA 解码程序进行优化后,提高了 WMA 的解码效率。优化结果可从两个方面来体现:首先, MPlayer 实时显示的 CPU 使用率已从 800%~1000% 降到 50%~100%。然后,通过对比播放同一个文件所用的时间来体现,如表 1 所示。假设:优化前的 MPlayer 生成的可执行文件为 `mplayer-1`,优化后的为 `mplayer-2`。`test1.wma` 是一个采样率为 44kHz,比特率为 48 kbps,长度为 20 s 的单通道 WMA 文件。`test2.wma` 是采样率为 44 kHz,比特率为 48 kbps,长度为 20 s 的一个双通道 WMA 文件。其中,解码耗时包含初始化时间(很短,可以近似不计)。

表 1 优化前后解码耗时对比

File	Player	Decoded Time/s
test1.wma	mplayer-1	53.73
test1.wma	mplayer-2	21.42
test2.wma	mplayer-1	338.59
test2.wma	mplayer-2	29.77

总之,优化后的 MPlayer 已经能在 S3C2410 平台上流畅播放单通道的 WMA 音频文件,而双通道的 WMA 音频文件还不是很流畅。尽管如此,相比没有优化的 WMA 解码库,解码效率已经得到了显著提高。

## 5 结束语

优化工作在嵌入式开发中扮演着重要的角色,高效的优化方案能显著地提高系统的性能<sup>[8]</sup>。在优化过程中,不仅要先分析系统中最需要优化的地方,优先对系统的瓶颈进行优化,还要根据目标平台的特性提出针对性的优化方案。只有牢牢把握住这两个方面,才能使优化工作收到立竿见影的成效。本文以这两个方面为基础,使用浮点运算转定点运算、以空间换时间以及手工汇编这 3 种方案对 MPlayer 中的 WMA 解码程序进行优化,显著地提高了其在 S3C2410 平台上的解码效率。

## 参考文献:

- [1] 傅立叶开发部. FFT2410 简易板使用说明书[M/CD]. 傅立叶电子科技有限公司, 2005.
- [2] MPlayer. The source code packages of MPlayer[EB/OL]. <http://www.mplayerhq.hu/MPlayer/releases/>, 2008-03-07.

(下转第 34 页)

## 5 结束语

本文所作的工作有以下几个方面:研究了隶属函数确立的基本原则、基本步骤和常用算法,在此基础上,论述了在IIDS系统中所用到的3种算法:一是采用非模糊统计法来建立隶属函数的表达式;二是利用遗传算法来优化隶属函数的参数,从而搜索到最优的隶属函数;三是采用改进后的层次分析法与熵值法相结合的综合分析法来确立因素的权重。

模糊控制技术是当今控制领域应用较为前沿和成熟的技术,将其理论和技术应用到入侵检测系统中,对于提高入侵预报的准确度、可信度以及增强入侵检测系统的辅助决策能力可起到很好的效果。相信通过模糊控制技术在网络安全领域的不断推广应用,将进一步提高入侵检测系统的智能化水平,从而进一步增强网络系统的安全性和可靠性,创造出更大的经济效益和社会效益。

## 参考文献:

- [1] 李开银.基于异常检测和误用检测的入侵检测系统[D].杭州:浙江大学,2004:1-5.
- [2] Axelsson S.The base-rate fallacy and the difficulty of intrusion detection[J].ACM Transaction Information and System Security, 2000,3(3):186-205.
- [3] 戴西超,张庆春.综合评价中权重系数确定方法的比较研究[J].煤炭经济研究,2003(11):15-17.
- [4] 孙东,黄天戌.基于模糊数据挖掘与遗传算法的异常检测方法[J].计算机应用,2006,26(1):66-69.
- [5] 王明涛.多指标综合评价中权重系数确定的一种综合分析方法[J].系统工程,1999(3):11-14.
- [6] 朱茵,孟志勇.用层次分析法计算权重[J].北方交通大学土木建筑工程学院,1999(10):25-29.
- [7] 张跃.模糊数学方法及其应用[M].北京:煤炭工业出版社,1992:88-98.
- [8] Long Zhiqiang,Lv Zhiguo,Hong Huajie.Realizing the fault diagnostic system in maglev suspension train system based on fuzzy comprehensive evaluation method[C].Proceedings of the 4th World Congress on Intelligent Control and Automation, 2002.
- [9] 段玉倩,贺家李.遗传算法及其改进[J].电力系统及其自动化学报,1998,10(1):39-52.
- [10] 鲁红英,肖思和.基于改进的遗传神经网络数据挖掘方法研究[J].计算机应用,2006,26(4):38-41.
- [11] 丁承民,张传生.遗传算法纵横谈[J].信息与控制,1997,26(1):40-48.
- [12] Srinvas M,Patnaik L M.Adaptive probability of crossover and mutation in genetic algorithms[J].IEEE Trans on SMC,1994,24(4):656-667.
- [13] 易伟民,申群泰.整流机组效率优化中遗传算法的研究与应用[J].计算机应用,2003,23(3):19-22.

(上接第 15 页)

- [3] MPlayer. The online documentation of MPlayer[EB/OL]. <http://www.mplayerhq.hu/DOCS/HTML-single/en/MPlayer.html>, 2008-04-01.
- [4] HYDE R. 编程卓越之道(第一卷) 深入理解计算机[M]. 韩东海,译.北京:电子工业出版社,2006:10-100.
- [5] Book of Hook. Introduction to fixed point math[EB/OL]. <http://trac.bookofhook.com/bookofhook/trac.cgi/wiki/IntroductionTo-FixedPointMath>, 2004-09-14.
- [6] YATES R. Fixed-point arithmetic: an introduction [EB/OL]. <http://www.digitalsignallabs.com/>,2007-08-23.
- [7] BARR M. Programming embedded systems[M].南京:东南大学出版社,2007:1-301.
- [8] 胡贯荣,谢争捷,涂刚.嵌入式音频系统的设计与实现[J].计算机工程与设计,2006,27(23):4566-4568.

(上接第 28 页)

- [8] Du W,Deng J,Han Y S,et al.A pairwise key pre-distribution scheme for wireless sensor networks[C]. Proceedings of 10th ACM Conference on Computer and Communications Security, 2003.
- [9] Liu D,Ning P.Establishing pairwise keys in distributed sensor networks[C]. Proceedings of 10th ACM Conference on Computer and Communications Security, 2003:52-61.
- [10] Traynor P.Establishing pairwise keys for heterogeneous sensor networks[R].Tech Rep,The PA State Univ, 2004.
- [11] Eltoweissy M, Moharram M, Mukkamala R.Dynamic key management in sensor networks[J]. IEEE Communications, 2006,44(4):122-130.
- [12] Yu Z,Guan Y.A key pre-distribution scheme using deployment knowledge for wireless sensor networks [C]. Proceedings of ACM/IEEE International Conference of Information Processing in Sensor Networks, 2005.